

# SYSTÈMES D'EXPLOITATION : LES PROCESSUS - EXERCICES

## PROCESSUS

### Exercice 1 : afficher la liste des processus

1. Ouvrir « powershell »
2. Sous Windows :
  - a) Afficher la liste des processus avec la commande `tasklist /V`
  - b) Retrouver le nom du tout premier processus lancé par le noyau : quel est son PID ?
  - c) Le soucis avec cet affichage, c'est que la liste des processus est longue, et qu'on ne voit pas l'entête des colonnes. Pour y remédier, compléter la commande `tasklist` ainsi :  
`tasklist /V | more`      *appuyer sur « espace » pour changer de page*
  - d) Trouver le processus « `tasklist` » :
    - Quel est son PID ?
    - Dans quel état est-il ? Pourquoi ?
3. Sous Linux:
  - a) Se connecter au serveur ssh
  - b) Afficher la liste des processus avec la commande `ps aux`
  - c) Retrouver le nom du tout premier processus lancé par le noyau : quel est son PID ?
  - d) Le soucis avec cet affichage, c'est que la liste des processus est longue, et qu'on ne voit pas l'entête des colonnes. Pour y remédier, compléter la commande `tasklist` ainsi :  
`ps aux | more`      *appuyer sur « espace » pour changer de page*
  - e) Trouver le processus « `ps` » :
    - Quel est son PID ?
    - Dans quel état est-il ? Pourquoi ?

**Pour la suite des exercices, si ça n'est pas encore fait, se connecter à votre compte lfcltx sur le serveur ssh avec la commande suivante dans un terminal :**

**> ssh lfcltx@lfcl.szpieg.fr** , en remplaçant x par votre numéro

**Dans votre home directory, créez un répertoire Processus**

### **Exercice 2 : liste des processus**

1. Une fois que tout le monde est connecté au serveur ssh, afficher la liste des processus avec la commande de l'exercice 1.
2. Repérer les processus utilisés par les autres élèves du groupe : grâce à quelle colonne peut-on le faire ?
3. Avec cette commande, il y a tous les processus du serveur. Pour sélectionner uniquement les processus utilisés par nous, on peut trier avec la commande suivante :  
    > ps aux | grep lfcl

### Exercice 3 : tuer un processus

1. Ouvrir un 2<sup>e</sup> terminal, et connectez-vous une deuxième fois au serveur ssh avec votre compte lfcltx.
2. a) Quelle commande permet de voir les utilisateurs connectés ?  
b) Taper cette commande pour vérifier que vous êtes bien connecté deux fois (vous devriez voir les autres élèves connectés deux fois également).
3. a) Dans un des deux terminaux, taper la commande « cat » puis validez. Cette commande attend que l'utilisateur écrive du texte au clavier.  
  
b) Dans l'autre terminal, grâce à la commande « ps aux » (+ les paramètres à trouver pour simplifier la recherche), retrouver le PID du processus généré par **votre** commande cat.  
  
c) Quel est l'état de ce processus ? Expliquer pourquoi.  
  
d) Vous allez tuer ce processus grâce à la commande suivante (dans le 2<sup>e</sup> terminal, en gardant la fenêtre du premier terminal visible) :  
    >kill numeroduprocessus  
  
e) Dans le premier terminal, que s'est-il passé ?  
  
f) Avec la commande « ps aux » (+ les paramètres à trouver pour simplifier la recherche), chercher le processus tué. Le trouve-t-on ? Pourquoi ?

## Exercice 4 : arbres

1. Garder les deux terminaux connectés au serveur ssh avec votre session.
2. a) Dans l'un d'eux, afficher l'arbre des processus avec la commande « pstree -up ».  
b) De quels utilisateurs voit-on les processus ?
3. Dans le 2<sup>e</sup> terminal, tapez la commande « cat ».
4. a) Dans le 1<sup>er</sup> terminal, rechercher le PID du processus généré par **votre** commande cat (voir question 3.b de l'exercice précédent)  
  
b) Afficher la liste de tous les ancêtres de ce processus avec la commande suivante :  
    > pstree -ups numeroduprocessus

*Remarque : « u » pour afficher les utilisateurs, « p » pour les PID et « s » pour « Show-parents », donc pour montrer les processus parents (au sens ancêtre) du processus en question.*

## INTERBLOCAGE

### Exercice 5 : accès concurrents à un répertoire partagé 1 (verrou)

Imaginons qu'un processus veuille écrire dans un fichier. Pendant cette opération, il ne veut pas qu'un autre processus puisse écrire en même temps que lui dans ce fichier. Pour éviter l'accès simultané de deux processus à une même ressource, le premier processus va fermer un verrou (*lock*).

Si un 2<sup>e</sup> processus veut écrire dans ce fichier, il va au préalable tenter de fermer le verrou. Celui-ci étant déjà fermé, il ne peut pas le faire et le noyau le met à l'état « suspendu ». Le noyau remettra le 2<sup>e</sup> processus à l'état prêt lorsque le processus 1<sup>er</sup> aura rouvert le verrou.

Mise en pratique : on va empêcher l'utilisation simultanée par tous les élèves de la commande « cat ». A chaque instant, un seul d'entre vous pourra utiliser cette commande.

Pour protéger cette commande, nous allons déposer un verrou sur le fichier /tmp/verrou, accessible à tous.

La ligne de commande est la suivante :

```
> flock /tmp/verrou -c cat
```

*Remarques : « flock » (File lock) permet de mettre un verrou sur un fichier (ici /tmp/verrou), et « -c » indique la commande à protéger (ici cat)*

Taper la commande, valider et attendre (**ne touchez plus à rien!!**)

Remarque : pour arrêter la commande « cat » qui attend une chaîne de caractère tapée au clavier, il faut appuyer sur Ctrl+d.

Remarque : la commande « cat » n'a été lancée que pour celui qui a fermé le verrou. Donc le terminal ne va réagir à Ctrl+d que pour cette personne.

## Exercice 6 : accès concurrents à une ressource partagée 2 (verrou)

1. Copier dans votre répertoire Processus les deux fichiers de script bash Process1.sh et Process2.sh qui se trouvent dans /home/lfcltpartage/Processus/
2.
  - a) Commentaires communs sur ces fichiers.
  - b) Lancer le 1<sup>er</sup> des deux fichiers avec la commande suivante :  
> ./Process1.sh votreprenom
  - c) Avec la commande « ls », rechercher le fichier créé par le programme
  - d) Avec la commande « cat », vérifier que le contenu du fichier est bien conforme à vos attentes.
  - e) Relancer la commande de la question 2.b), que se passe-t-il dans ce fichier ?
  - f) Pour la suite de l'exercice, effacer le fichier grâce à la commande « rm ». Vérifier avec la commande « ls » que le fichier a bien été supprimé.
3. Si ça n'est pas déjà le cas, ouvrir un 2<sup>e</sup> terminal et se connectez une 2<sup>e</sup> fois sur le serveur ssh. Dans les deux terminaux, se placer dans le répertoire Processus de votre Home directory.  
Vous pouvez vérifier avec la commande « ls » que les deux fichiers de la question 1 sont présents dans ce répertoire.
4. Lancer dans un terminal le premier script Processus1.sh avec la commande suivante :  
> ./Process1.sh votreprenom  
Dans les 15 secondes qui suivent, faire la même chose avec dans le 2<sup>e</sup> terminal avec le 2<sup>e</sup> script :  
> ./Process2.sh votreprenom  
Observer simultanément les deux terminaux pendant l'exécution des deux scripts : que constate-t-on ?
5. Afficher le contenu du fichier R1.txt. Expliquer pourquoi les lignes écrites par les scripts 1 et 2 ne sont pas mélangées.

## Exercice 7 : interblocage

1. Copiez dans votre répertoire Processus les deux fichiers de script bash Interb1.sh et Interb2.sh. qui se trouvent dans /home/lfcltpartage/Processus/
2. Commentaires communs sur ces fichiers.
3. Si ça n'est pas déjà le cas, ouvrez un 2<sup>e</sup> puis un 3<sup>e</sup> terminal et connectez-vous à chaque fois sur le serveur ssh. Dans deux terminaux, placez vous dans le répertoire Processus de votre Home directory.  
Vous pouvez vérifier avec la commande « ls » que les deux fichiers de la question 1 sont présents dans ce répertoire.
4. a) Lancez dans un terminal le premier script Interb1.sh avec la commande suivante :  
    > ./Interb1.sh  
  
b) Dans les 15 secondes qui suivent, faites la même chose dans le 2<sup>e</sup> terminal avec le 2<sup>e</sup> script :  
    > ./Interb2.sh  
  
c) Attendre : que se passe-t-il ? Pourquoi ?
5. Dans le 3<sup>e</sup> terminal :
  - a) Accédez aux verrous avec la commande suivante :  
    > lsof | grep R2.txt
  - b) Dessiner l'arbre (restreint) des processus des scripts 1 et 2 avec la commande suivante :  
    > pstree -sup xxxxx , où xxxxx est le numéro du PID du processus du script 1 ; puis refaire la même chose avec le script 2.
  - c) Quel(s) processus peut-on tuer pour débloquent l'interblocage ? Le faire après mon autorisation !
  - d) Vérifier que les scripts se débloquent bien dans l'ordre prévu.

## **Exercice 8 : garder une trace de son travail**

**Avant de vous déconnecter**, tapez la commande suivante :

```
> history>/home/lfcltx/Processus/Processhist.txt.
```

Si vous voulez aller voir toutes les commandes tapées sur ce TP, il suffira de taper :

```
> cat /home/lfcltx/Processus/Processhist.txt.
```

Pour quitter le serveur « lfcl.szpieg.fr » tapez « exit » (à faire pour éviter de faire tourner la session « pour rien » svp).